An online framework for state of charge determination of battery systems using combined system identification approach

Mohammad Rezwan Khan [a,b,∗], Grietus Mulder [b], Joeri Van Mierlo [a]

[a] Vrije Universiteit Brussel, Mobility and Automotive Technology Research Group, Brussels, Belgium
[b] Vlaamse Instelling voor Technologisch Onderzoek, Unit Energy Technology, Mol, Belgium

HIGHLIGHTS

- Use of two system identification optimization (Genetic Algorithm and Levenberg-Marquardt).
- Algorithms are adapted for extracting parameter of batteries.
- Use of novel error function based hysteresis compensation.
- Newly developed online state of charge estimation framework.
- Designed for most battery chemistries with minimal adaptation.

GRAPHICAL ABSTRACT
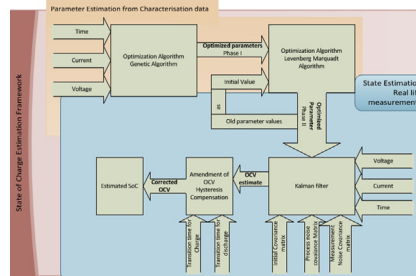
ABSTRACT

In this article, an online state of charge (SoC) estimation framework is proposed, designed and implemented using the system identification approaches. The techniques are composed of cross combination between two modified nonlinear optimisation algorithms (modified Genetic Algorithm and modified Levenberg Marquardt) adapted for battery cell parameter estimation. Subsequently a linear recursive Kalman filter is employed to estimate the state parameters of the battery cell. Moreover, a newly statistical approach is developed to encounter hysteresis phenomenon within the cell. The prerequisite for the SoC determination in the electrical vehicle (EV) is challenging as the battery can be composed of hundreds of cells while the load current changes dramatically inside the cells and the required elapsed time for SoC determination should be as short as possible to extend the expected lifetime of the battery pack. Thus, the accurate estimation of the SoC of the cells in a battery pack is one of the key factors for using them effectively. The framework is found to be robust, optimal and implementable in time constrained environment with acceptable accuracy.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The state of charge (SoC) of a battery is a measure of the amount of electrical energy stored in the battery. An accurate SoC estimation depends on two aspects according the definition of SoC given by coulomb counting (Ah) equation: one is the initial SoC (S_0), coulombic efficiency (η), and the other is the calculation of SoC consumption,

$$S_t = S_0 - \int_0^t \frac{\eta I}{C_n} dt \tag{1}$$

So in case that the initial state of charge and the actual true capacity (C_n) with coulombic efficiency is known, Eq. (1) provides

the quickest and the most precise value of the state of charge of the battery cells. Table 1 provides an inventory of technical terms used in the article. In reality due to the variability among the cells and the limitation of real life sensor accuracy, it is impossible to determine the true capacities for each cell in a battery pack and the initial state of charge is also very difficult to determine confidently due to different real life battery cell characteristics e.g. coulombic over potential [1]. Furthermore, because of the nonlinear behaviour of the cells inside a battery pack and the absence of a definitive model to interpret the behaviour of each cell state of charge determination can be a challenging task. The problem is exacerbated in case of a battery pack operational in EVs or grid level that contains a number of cells (e.g. Opel Ampera: 288 cells, Nissan Leaf: 196 cells) and the load current is highly varying according to the driving profile. It is the intention to find the state of charge of different cells at once in an online[1] automated manner. Especially for the online battery state of charge determination system, the task requires a solution for a number of challenges in the field of modelling which consists of the identification of the parameters, (near) accurate mathematical representation, assessing deterministic or stochastic behaviour of the system, and ultimately translating all these pieces of the information into one integrated setting that can evolve with the change of the surroundings which can eventually validate the utility of the methodology.

The SoC of the battery is a complex non-linear function of parameters. Key factors affecting the SoC are the charge–discharge rates, hysteresis, temperature, cell age, and self-discharge [1]. The existing SoC estimation methods can generally be categorised into two types: direct kind and indirect kind schemes. One of the direct approaches simply indicates the remaining capacity by using current integration or coulomb counting technique. However, a large SoC estimation error exists caused by the inaccurate initial SoC and the low calculation accuracy of the coulomb (Ah) consumption due to some uncertain disturbances originating from the real life sensor characteristics. On the other hand, the indirect methods determine the SoC by using the battery's intrinsic relationship between the SoC and some electrical parameters such as open circuit voltage (OCV). OCV naturally declines proportionately with the energy expenditure and is widely used for SoC estimation. But the direct measurement of the OCV is difficult for online application due to the long waiting time for the batteries to reach a steady state condition [2,3]. A variety of techniques has been proposed to measure or monitor the SoC of a cell or battery each has its relative merits, as reviewed by Ref. [4]. Some methods are specific to particular cell chemistries. Most methods depend on measuring some convenient parameter, which varies with the SoC. [5] proposed a real-time SoC evaluation system for Li-ion batteries combining direct measurement of electromotive force (EMF) during the equilibrium state and coulomb counting during the charge–discharge states. [6] implemented a sliding-mode observer for robust tracking under nonlinear conditions based on a RC battery model. [7] proposed a SoC determination method based on an extended Kalman filter for observation of the parameters of the modified Randles circuit battery model. [8] used a recursive least square algorithm with an optimal forgetting factor for estimating OCV from the battery under test using an OCV–SoC look up table. [9] used a Kalman filter to find the state of charge of batteries and did not take account of hysteresis effect which diminishes the usage of the direct application of their system to estimate SoC for most of the batteries. [10] used two extended Kalman filter (EKF) loops to integrate the hysteresis phenomenon, one for SoC estimation and

one to offset the hysteresis effect. The usage of two EKF makes the system computationally burdened. Also both of [9,10] use an experimental method to derive the parameters from the batteries without further optimisation and for that reason there is a chance of failure to represent the best dynamic battery behaviour as these values did not provide the best fit to the terminal values since parameter optimisation was not done to get the best fit.

This article proposes a new framework for the state of charge estimation considering the use of two different nonlinear optimisation algorithms namely modified Genetic Algorithm (Section 3.3.1) and modified Levenberg Marquardt method (Section 3.3.2) for parameter estimation and using Kalman filter (Section 3.4) for battery state estimation. This article addresses the hysteresis concern of the OCV curve of battery cells with the introduction of a probabilistic parameter (Section 3.5). Overall, this state of charge estimation method is designed for the required purpose of SoC determination in varying dynamic conditions of large battery packs, e.g., hybrid electric vehicle (HEV), Battery electric vehicle (BEV) or grid scale battery pack that operates in dynamic environments.

2. Methodology

The fundamental part of the proposed framework is a coupling of the estimation methodologies for the state and parameter estimation. The framework is illustrated in Fig. 1. A lumped parameter battery model is used to estimate different voltage levels and this

Table 1
Different technical terms and respective definition.

Terms	Definition
C-rate	The rate at which a battery can deliver or accept current, stated in terms of the rated capacity of the cell in amp-hours. This may also be referred to as the hour rate, such as the 1-h rate.
Coulombic efficiency, η	Ratio of charge delivered by a rechargeable battery during discharge cycle to the charge stored during charge cycle.
Battery Management System (BMS)	An electronic device assembly or system that monitors and controls a rechargeable battery. Parameters measured by the system may include cell temperature, voltage, and current. From this data, the BMS can compute the state of charge of the battery and other necessary battery state. The BMS may also contain sensors and circuitry for protection such as over-current, over-temperature, or over-voltage.
Battery pack	Collection of batteries or individual cells electrically connected in series and/or parallel combinations along with the required electrical interconnections, mechanical packaging, thermal management, and sensing circuitry. Because it is a self-contained assembly, a battery pack can be easily swapped in and out of the application.
SoC	Defined as the capacity left in a battery expressed as a percentage of some reference. SoC of a battery is usually expressed as a percentage of the current battery capacity when it is fully charged.
Initial state of charge S_o	Battery Starting SoC condition for a defined application. Full battery means 100% SoC while 0% means Empty battery.
True capacity, C_n	True or nominal capacity is different from the physical or theoretical capacity of a cell, which is dependent upon the active material in the cell and is based upon the total amount of energy that can be stored or extracted from the cell when manufactured. It can be reached by discharging the cell with a very small current at the specified temperature.
Coulombic capacity, Ah	Unit of electric charge, the amount of current delivered for 1 h

---

[1] Online: The processing of data piece-by-piece in a sequential manner without availability of the entire data from the start of estimation.
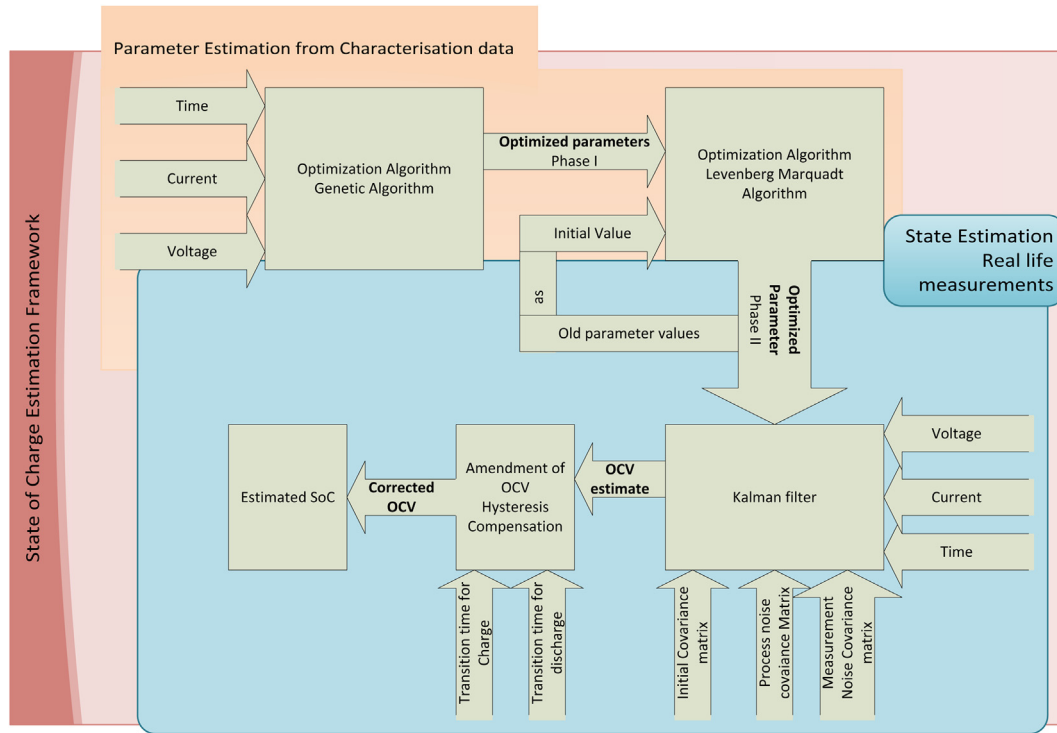
Fig. 1. Overview of the online SoC Estimation framework.

voltage states estimation is carried out after the execution of a Kalman filter. The filter output consists of the estimated terminal voltage of the battery and a modified OCV (the voltage that is used with hysteresis curve to determine SoC). The article addresses the hysteresis concern with the introduction of a probabilistic parameter by tracking consecutive charge and discharge loads since most battery cells exhibit hysteresis and it should be included in order to determine correctly the state of charge of the battery. The following sections elaborate the sequential steps behind the online state of charge estimation in the proposed framework. The presented article describes the application of state-estimation techniques for the online prediction of the state-of-charge of Li-ion cells. Traditional and commonly used coulomb-counting techniques unfortunately suffer from offset, drift, and long-term state divergence originating from real life sensor characteristics [11]. To provide adjustment for these concerns a modified open circuit voltage (OCV) is used to estimate the SoC. The essential dynamic behaviour of each cell is modelled using two capacitors (bulk and surface) and three resistors (terminal, surface, and end). The SoC is determined from the voltage present on the bulk capacitor [9,12]. This RC equivalent circuit model (Fig. 2) is one of the familiar choices for the representation of a battery cell because of its ability to demonstrate dynamic characteristics [9,10]. In this article a load profile is used as
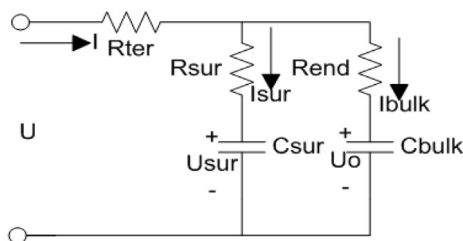
it is specified in the IEC 62660-1 standard for BEVs [13] with an adjustment. The modification is made to accommodate the full charging and discharging of the full battery cell. Without loss of oversimplification, the system has two types of member states (voltages on capacitances and terminal voltage) and parameters (resistance and capacitance values). In this article, the battery cell parameters are calculated offline[2] and states are calculated online. At the parameter estimation step, the optimisation routine executes on the *integrated pulse test* (IPT) [13] characterisation data where prior parameter estimates are calculated. It is notable that the error covariance is computed and the uncertainties in the parameters are not taken into account in the parameter estimation step. The parameter estimation step is composed of two sequential non-linear optimisation methods: Modified Genetic Algorithm and modified Levenberg Marquardt method in order to locate the optimum parameter set and thereby finding the minimum of the cost function of the algorithm properly. Subsequently, the online state estimation is carried out with the help of the optimal filter technique while taking account of the uncertainty originated from the model and the measurement. The implementation of the integration of the unpredictable process noise or uncertainty and the measurement noise diminishes the error in case of the parameter estimates through an optimal filtering technique like Kalman filter. To get the SoC from the battery state voltage, a SoC–OCV look-up table is used. A new aspect is the inclusion of hysteresis and the deduction of the table, that will explained further on.

3. Framework formation

The parameterisation process is initiated by building the state-space model representation of the RC equivalent model representing a battery cell (Fig. 2). The parameters are sought that result

Fig. 2. RC battery equivalent model.

into a calculated voltage profile that is the closest to the measured profile. Non-linear optimisation algorithms are used and will be explained in the next sections.

3.1. State-space representation of equivalent RC model

The equivalent model that represents the static and dynamic behaviour of the battery should be wisely constructed to track the state of charge of the battery. When selecting a model, the aim is to find the most parsimonious model that adequately describes real life battery behaviour. A simple model is easier to estimate, forecast, and interpret but has sometimes the weakness of failure to demonstrate the dynamic behaviour [11]. The single order equivalent battery cannot realistically simulate the non-linear dynamic behaviour of the battery system due to the simplified dynamic characteristics. On the other hand, a higher order battery model is sometimes difficult to implement and operate in the time constrained environment. The strategy is to use a less complex model that is sufficient enough to represent the data and the measurement noise model is used to avoid errors between the battery system and model. This paradigm can serve to offset the effect of the simplified model and the usage of KF algorithm ensures reducing measurement errors [14,15]. The RC equivalent model is a dynamic model of the battery which consists of a bulk capacitor $-C_{bulk}$ to characterise the ability of the battery to store charge, a capacitor to model surface capacitance and diffusion effects within the battery $- C_{sur}$, a terminal resistance $- R_{ter}$, surface resistance $- R_{sur}$, and end resistance $- R_{end}$ [12] as shown in Fig. 2. The voltage across the bulk and surface capacitors are denoted U_o and U_{sur} respectively. The parameters of the cell are extracted from experimental pulse data using optimisation techniques, where IPT tests are performed upon successive charge and discharges of the battery cell under test. A dynamic model of the battery, in the form of state variable equations, is necessary to predict the SoC of the cell using KF [9].

3.2. Parameterisation

3.2.1. Parameterisation data requirements

Due to the variability in the voltage level observed in the battery charge and discharge curves, a set of different rate of charge and discharge curves (IPT data) are used for the parameter tuning process to diminish the variability. The quality and consistency of the lab data is very important in achieving a good fit in this article's parameter estimation step. Uncertainty and real life sensor characteristics are assumed to have no impact in the parameter estimation step. The parameter estimation data also show some significant inconsistency between the tests on different temperatures of the battery cell. For this reason the experiment is conducted in the isothermal environment.

3.2.2. Experiments for parameterisation

The configuration of the battery test bench is shown in Fig. 3 at the right and a 12 Ah Nickel Manganese Cobalt Oxide (NMC) based pouch shaped Li-ion battery used as one of the cells under test is shown at the left. The test equipment is the PEC™ SBT0550 cell tester. In order to limit the temperature's influence on the model parameters, all of the experiments of the Li-ion NMC battery cell are carried out in a thermal chamber with a fixed temperature of 20 °C. To identify the model parameters of the battery cell, the integrated pulse test (IPT) [13] is performed. The integrated pulse test is based primarily on the power density test from the IEC 62660-1 standard [16] and the HPPC test from the US Department of Energy battery test manual for (plug-in) hybrid electric vehicles [17]. The IPT test consists of a series of several discharge and charge pulses with a charge sustained operation that are executed at different SoC levels: 80, 65, 50, 35 and 20%. After the pulse series at a definite SoC level, the cell is discharged to the next SoC level. The rest period between the discharge pulses is restricted to 40 s. The rest period after a charge pulse is 10 min. Between the pulse trains there is a half hour

$$\frac{d}{dt}\begin{pmatrix} U_0 \\ U_{sur} \\ U \end{pmatrix} = \begin{pmatrix} \frac{-1}{C_{bulk}(R_{end}+R_{sur})} & \frac{1}{C_{bulk}(R_{end}+R_{sur})} & 0 \\ \frac{1}{C_{sur}(R_{end}+R_{sur})} & \frac{-1}{C_{sur}(R_{end}+R_{sur})} & 0 \\ A_{31} & 0 & A_{33} \end{pmatrix}\begin{pmatrix} U_0 \\ U_{sur} \\ U \end{pmatrix} + \begin{pmatrix} \frac{R_{sur}}{C_{bulk}(R_{end}+R_{sur})} \\ \frac{R_{end}}{C_{sur}(R_{end}+R_{sur})} \\ \frac{R_{end}^2}{C_{sur}(R_{end}+R_{sur})^2} - \frac{R_{sur}R_{ter}}{C_{bulk}R_{end}(R_{end}+R_{sur})} + \frac{R_{ter}}{C_{sur}(R_{end}+R_{sur})} + \frac{R_{end}R_{sur}}{C_{sur}(R_{end}+R_{sur})^2} \end{pmatrix}(I) \quad (2)$$

where,

$$A_{31} = \frac{-R_{sur}}{C_{bulk}(R_{end}+R_{sur})^2} + \frac{R_{end}}{C_{sur}(R_{end}+R_{sur})^2}$$
$$- \frac{R_{sur}^2}{C_{bulk}R_{end}(R_{end}+R_{sur})^2} + \frac{R_{sur}}{C_{sur}(R_{end}+R_{sur})^2}$$

and

$$A_{33} = \frac{R_{sur}}{C_{bulk}R_{end}(R_{end}+R_{sur})} - \frac{1}{C_{sur}(R_{end}+R_{sur})}$$

Output equation,

$$U = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} U_0 \\ U_{sur} \\ U \end{pmatrix} \quad (3)$$

rest. The pulses of 10 s each have the following amplitudes: $0.33I_t$, $1I_t$, $2I_t$; $5I_t$; $10I_t$; $20I_t$ (I_t is the corresponding current the 1C capacity); maximum charge current and maximum discharge current. If the maximum allowed charge pulse is lower than the discharge pulse, the charge duration is increased proportionately to maintain the same SoC level, the so-called charge sustaining operation.

3.3. Parameterisation algorithms

The parameters of this battery cell are needed to have an approximate solution of battery systems to represent the cells in battery pack behaviour to estimate SoC accurately. The following sections represent a brief synopsis of the parameter estimation method:

3.3.1. Modified Genetic Algorithm

The Genetic Algorithm (GA) is a heuristic method for solving both constrained and unconstrained optimisation problems based

Fig. 3. [Left] Nickel manganese cobalt oxide based 12 Ah cell and [right] testing facility in PEC SBT0550 cell tester.

on natural selection; the process that mimics biological evolution. GA frequently modifies a population of individual solutions. At each step, GA selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. In the process, solutions which are not optimal tend to perish while other fitter solutions mate and propagate their desirable traits (survival of the fittest criterion), thus leading to more coherent solutions into the set that claim greater potential (the total set size remains constant; for each new solution added, an old one is removed). Further, a small randomisation is added which makes sure that a set won't stagnate and simply fill up with numerous copies of the same solution. GA is likely to work better than traditional optimisation algorithms because it takes advantage of an entire set of solutions in the solution space instead of making use of single-point transition rules to move from one single instance in the solution space to another since that may get tangled in local extremes [18]. Eventually, after a number of iterations, i.e. over successive generations, the population "evolves" toward an optimal solution. Application of GA is found for solving a multiplicity of optimisation problems that are not well suited for the standard optimisation algorithms, including problems in which the objective function is discontinuous, non-differentiable, stochastic, or highly nonlinear as applicable in case of this battery parameterisation. Unlike classical algorithms like steepest descent or Gauss-Newton, at each iteration GA generates a population of numerous points. The best point in the population approaches an optimal solution, whereas classical algorithms generate a single point at that iteration. The sequence of points in GA approaches an optimal solution. Instead GA selects the next population by computation which uses random number generators, whereas classical algorithm selects the next point in the sequence by a deterministic computation.

An overview of the steps in the Genetic Algorithm optimisation is shown in Table 2. In Table 3 an example is given of the evolution of chromosomes. The GA algorithm begins by creating a random initial population. The population size signifies how many chromosomes are in population (in one generation). The chromosome represents an assumption value for the model [R_{end}, R_{sur}, R_{ter}, C_{bulk}, C_{sur}] like shown in Table 3. If there are too few chromosomes, GA has a few possibilities to perform crossover and only a small part of search space is explored. On the other hand, if there are too many chromosomes, GA slows down. Research shows that after some limit (which depends mainly on the encoding of the problem) it is

not useful to increase the population size, because it does not make solving the problem faster [19]. In this optimisation case the population size is chosen to be 120 as this number is a trade-off choice for speed of the convergence and the explorability of the search area or space. The algorithm then creates a sequence of new populations. To create the new population, the algorithm performs the following steps: Children are produced either by making random changes to a single parent—as set by mutation parameter—or by combining the vector entries of a pair of parents—as set by crossover parameter. Crossover probability says how often crossover is to be performed. If there is no crossover, offspring is an exact copy of parents. Crossover is made in hope that new chromosomes have good parts of old chromosomes and may be the new chromosomes are better. Subsequently, there is mutation, which is described by a mutation parameter and a mutation probability. In this work the mutation parameter is chosen to be 0.5 because that gives 50% survival probability of current population. The mutation probability says how often a part of a chromosome is mutated. If there is no mutation, offspring is taken after crossover (or copy) without any change. If mutation is performed, part of the chromosome is changed. Mutation is made to prevent GA falling into a local extreme, but it should not occur very often, because then GA is in fact changed to random search. In this work, the mutation probability is chosen to be 0.3 as this is to ensure slow escape from the local extreme while it explore the global extreme.

To make sure that Genetic Algorithms work effectively, a few criteria must be met:

Table 2
Overview of the steps in Genetic Algorithm optimization steps.

Genetic Algorithm
• Scores each member of the current population by computing its fitness value. In this work the fitness function is given by (Eq. (4)).
• Scales the raw fitness scores to convert them into a more usable range of values.
• Selects members, called parents, based on their fitness. Some of the individuals in the current population that have lower fitness are chosen as elite. These elite individuals are passed to the next population. The number of elite population is chosen to be 4 in the article.
• Produces children from the parents.
• Replaces the current population with the children to form the next generation. The algorithm stops when one of the stopping criteria is met.

Table 3
An example of chromosome evolution at different generation that provide the best fit in the Genetic Algorithm.

Step	Generation	$\chi_k^{(g)}$	R_{end} (mΩ)	R_{sur} (mΩ)	R_{ter} (mΩ)	C_{bulk} (F)	C_{sur} (F)	Fit (%)
1	2	110	4.0896	6.012255	5.379275	125806.7	217.041	56.31
2	4	87	4.0896	6.012255	5.379275	125806.7	217.041	58.04
3	6	78	1.1012	23.64673	5.736231	50624.74	190.0416	59.61
4	8	52	1.1012	23.64673	5.736231	50624.74	190.0416	61.12
5	10	91	1.1012	23.64673	5.736231	50624.74	190.0416	61.91
6	12	42	1.1012	23.64673	5.736231	50624.74	190.0416	63.64
7	14	79	4.3490	2.711327	3.175169	53646.09	301.0573	64.19
8	16	94	5.5940	7.651067	1.754779	46739.52	228.9734	64.89
9	18	87	1.3287	2.548555	4.816261	48135.33	213.5607	65.17
10	20	86	1.3287	2.548555	4.816261	48135.33	213.5607	66.17
11	22	22	1.3287	2.548555	4.816261	48135.33	213.5607	67.06
12	24	18	1.3287	2.548555	4.816261	48135.33	213.5607	67.001
13	26	66	1.3287	2.548555	4.816261	48135.33	213.5607	69.221
14	28	75	1.3287	2.548555	4.816261	48135.33	213.5607	71.812
15	30	78	1.3287	2.548555	4.816261	48135.33	213.5607	74.154
16	32	79	1.3287	2.548555	4.816261	48135.33	213.5607	76.125
17	34	98	1.3287	2.548555	4.816261	48135.33	213.5607	78.013
18	36	95	0.4212	1.265056	4.795774	49253.76	257.0176	79.12
19	38	57	0.4212	1.265056	4.795774	49253.76	257.0176	81.29
20	40	63	0.4212	1.265056	4.795774	49253.76	257.0176	82.07
21	42	71	0.4212	1.265056	4.795774	49253.76	257.0176	84.43
22	44	57	0.4212	1.265056	4.795774	49253.76	257.0176	89.15
23	46	32	0.4212	1.265056	4.795774	49253.76	257.0176	86.74
24	48	48	0.1648	3.853546	5.097981	48552.19	58.74669	89.87
25	50	87	2.6150	2.370724	3.165559	49938.71	38.00201	94.17
26	52	97	2.6150	2.370724	3.165559	49938.71	38.00201	94.185
27	54	67	2.6150	2.370724	3.165559	49938.71	38.00201	95.198
28	56	89	2.6150	2.370724	3.165559	49938.71	38.00201	96.198
29	58	95	2.6150	2.370724	3.165559	49938.71	38.00201	96.764
30	60	64	2.6150	2.370724	3.165559	49938.71	38.00201	96.897
31	62	115	2.6150	2.370724	3.165559	49938.71	38.00201	96.956
32	64	5	2.6150	2.370724	3.165559	49938.71	38.00201	97.14
33	66	87	2.6150	2.370724	3.165559	49938.71	38.00201	97.37
34	68	71	2.6150	2.370724	3.165559	49938.71	38.00201	97.62
35	70	73	2.6150	2.370724	3.165559	49938.71	38.00201	97.64
36	72	56	2.6150	2.370724	3.165559	49938.71	38.00201	98.07
37	74	65	2.6150	2.370724	3.165559	49938.71	38.00201	98.36
38	76	91	2.6150	2.370724	3.165559	49938.71	38.00201	98.58
39	78	87	2.6150	2.370724	3.165559	49938.71	38.00201	98.64
40	80	104	2	1.4	1.5	45020.34	45.93	98.71

○ It should be easy to evaluate how "good" a potential solution is relative to other potential solutions i.e. has a good fitness function.

○ It should be easy to segment the potential solutions that can vary independently.

GA gives a solution which is close to the absolute answer to find the global extremes [18,20]. An indication of the parameterisation using only GA can be found in Table 4.

The major modification for the GA is to have a better fitness function that is adopted for the battery parameter estimation. The selection of a good fitness function is an essential part for getting the best optimisation of the algorithm system in a timely manner. The classical fitting system normally follows stationary values but misses the peaks as the peak span can be short in time interval (see Fig. 4): the normal GA does not take the short peaks in to account. For improving the issue, a bias toward peak has to be introduced using a weight term. The following Eq. (4) represents the fitness function for the Modified Genetic Algorithm that is used to minimise:

$$\begin{cases} \min\left(\chi_k^{(g)}\right) \\ \left(\chi_k^{(g)}\right) = \sum_{k=1}^{N} \left(1 + \beta(\partial V_k)^n\right)\left(V_k - U\left(\chi_k^{(g)}\right)\right)^2 \end{cases} \quad (4)$$

∂V_k stands for the voltage gradient at sample k. $(1 + \beta(\partial V_k)^n)$ is a weight term which can be adjusted to follow peaks. The power term n provokes a bias for slopes, and thus for the pulses. The multiplication factor β is an amplifying factor of the weight. The higher n is chosen, the better the peak pulse fitting becomes. In the presented case $\beta = 3$ and $n = 4$ is used. $\chi_k^{(g)}$ is the estimated value of chromosome k at generation g (see Table 3). U is the estimated profile of terminal voltage, $U(\chi_k^{(g)})$ is the terminal voltage profile of the population χ at the individual k at generation g; N is the estimation length i.e. data length of total estimation. The selection of the optimisation algorithm control parameters (e.g. crossover and mutation parameters in modified GA, the maximum number of iteration and tolerance in modified Levenberg Marquardt method) for the battery parameterisation is a painstaking and protracted process. Determining the optimum settings for the estimation and simultaneously selecting which parameters to estimate is not a trivial task. Although the parameter estimation framework has knowledge of the model output terminal voltage values, it cannot

Table 4
Optimised parameter values after parameter estimation step.

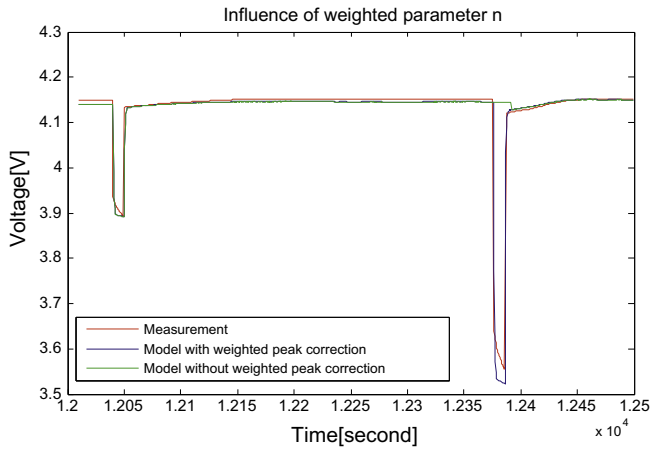| Method	R_{end} (mΩ)	R_{sur} (mΩ)	R_{ter} (mΩ)	C_{bulk} (F)	C_{sur} (F)
Genetic Alg.	8	3	2	38900	51.98
Modified Genetic Algorithm.	2	1.4	1.72	45020.34	45.93
Modified Levenberg Marquardt.	1.9	2.4	1.5	47643.66	46.96

Fig. 4. The figure illustrates the necessity of the bias parameter *n* for peak detection in battery parameter estimation stage. The measured value has large peaks for relatively low duration. From classical parameter estimated fitted data fails to catch the small peaks while the fitted peaks with weighted peak correction can fit among the peaks.

easily account for the intrinsic relationships between the parameters that are being tuned altogether. If multiple parameters being estimated have a similar effect on the output, the optimisation algorithms alone cannot distinguish between the similar parameters. For example, the ohmic voltage drop is influenced by three resistor components in the model.

Unless a sufficient variety of data is available to exercise all the estimated parameters fully, it is necessary to take a careful approach to estimate the parameter values. The usage of IPT data is motivated by the fact of probable non-orthogonal relationship among the parameters in all charge−discharge tests taken at different unique combinations of the operating conditions (the test runs at one temperature but with variable charge−discharge current rates applied at several SoC levels). Also, some learnt empirical reasoning has been made for the extraction of resistance and capacitance parameters through optimisation. The resistance and capacitance parameters are chosen to have non-zero and non-negative values. The initial lower and upper population bound is to be selected carefully. The resistance values in the model is quite important to estimate and the value behind the resistance is given in few mili-ohm range (lower bound value $0.1 \cdot 10^{-3}$ and upper bound value on $50 \cdot 10^{-3}$). The maximum bound value is motivated by the assumed fact that the resistance parameters are not expected to exceed 50 mili-ohm for this specific battery. For the bulk capacitance value (C_{bulk}) the lower bound is given $3 \cdot 10^4$ (assuming the battery as a large linear capacitor as proposed in Ref. [12]) and the highest bound value is not set. For surface capacitance (C_{sur}) the initial value is set 0.1% of the bulk capacitance (C_{bulk}) value from empirical observation. It is notable that these input boundary values reduce the significant computing power needed for the nonlinear optimisation as they have to search in a relatively small search range. Therefore, it ensures the speed of convergence to proper value.

3.3.2. Modified Levenberg-Marquardt method

Because of using two-tier parameter estimation system assures a more optimised version of the parameters, this concept has been used. The modified Genetic Algorithm is used to approximate the global minima for the nonlinear problem due to its ability to approximate the global extremes [21]. Afterwards the output of the modified GA (the best population) is fed into the modified Levenberg Marquardt [18] method to approximate the local minima in the proximity of the global minima for finding

more precise global extremes value of the solution space. The Levenberg Marquardt (LM) algorithm is an iterative technique that locates the minimum of a nonlinear function. It can be thought of as a combination of steepest descent and the Gauss-Newton method. When the current solution is far from the correct one, the algorithm behaves like a steepest descent method: slow but guaranteed to converge. When the current solution is close to the correct solutions, it becomes a Gauss-Newton method [18].

Let, *f* be an assumed functional relation which maps a parameter vector $\theta \varepsilon \Re^m$ to an estimated measurement vector $\widehat{x} = f(\theta);$, $\widehat{x} \varepsilon \Re^n$ and $\theta = [R_{end}, R_{sur}, R_{ter}, C_{bulk} \; C_{sur}]$. Table 5 gives an example of the LM method. An initial parameter estimate θ_0 which is the optimised population that comes from the output of previous modified GA stage and a measured vector *x* (the same measurement data that are used for modified GA) are provided and it is desired to locate the vector θ^+ (the most optimised parameter steps) that satisfy the best functional relation of the model *f*, i.e. minimizes the provided fitness function. Like all non-linear optimisation methods, LM is iterative: Initiated at the starting point θ_0, the method produces a series of vectors $\theta_1, \theta_2, \ldots$ that converges towards a local minimize θ^+ for desired function *f* [22,23]. In this way, LM can defensively navigate a region of the parameter space in which the model is highly nonlinear. A LM step approximates the exact quadratic step appropriate for a fully linear problem. LM is adaptive because it controls its own damping (The strategy of altering the diagonal elements of the approximate hessian $J^T J$ is called damping, *J* is the Jacobian of the parameter values): it raises the damping if a step fails to reduce error; otherwise it reduces the damping. The LM algorithm terminates when at least one of the following conditions is met:

○ The magnitude of the error gradient drops below a threshold.
○ The error drops below a predefined threshold. In the corresponding case 1.10^{-6} is used as Termination of tolerance function value.
○ A maximum number of iterations are completed. In this article number of maximum iteration is taken equal to 1200.

Algorithm pseudo code can be found in Ref. [24]. The major modification is incorporated in the goodness function. The modification is to make the goodness function more sensitive to short peaks like used in the modified GA case as stated by Eq. (4). The lower bounds are provided with the same output as found from the execution of modified GA. The upper bound for the algorithm is not set.

Table 5
An example of the evolution the parameters in the Levenberg-Marquardt method.

Step	Θ	R_{end} (mΩ)	R_{sur} (mΩ)	R_{ter} (mΩ)	C_{bulk} (F)	C_{sur} (F)	Error (%)
1	θ_1	2	1.4	1.72	45020.3	45.93	1.288742
41	θ_{41}	2.8	1.7	1.8	41700.4	42.01	1.284725
81	θ_{81}	3	3.2	1.9	38900	40.67	1.278179
121	θ_{121}	3.2	3.6	2	37000	39.74	1.246885
161	θ_{161}	4	3.8	2.1	36000	39.67	1.22321
201	θ_{201}	6	3.3	2.25	31000	34.78	1.222814
241	θ_{241}	6	4	2.4	31000	33.54	1.186281
281	θ_{281}	3	3	1.7	41700	42.01	1.156034
321	θ_{321}	2.1	3.6	1.8	42000	40.67	1.128643
361	θ_{361}	2.3	3.3	1.55	40000	39.74	1.127941
401	θ_{401}	1.6	3.3	1.45	45000	39.67	1.093907
441	θ_{441}	1.65	3.2	1.25	47000	46.09	1.05554
481	θ_{481}	1.7	3.1	1.61	47600	46.47	1.052537
521	θ_{521}	1.4	2.4	1.51	48000	47.05	1.024115
561	θ_{561}	1.9	2.4	1.5	47643.7	46.96	1.000565

3.4. State of charge estimation using Kalman filter

If the SoC estimation only depends on the model then it may be vulnerable to different uncertainties coming from the model as well as the characteristics of the real life sensors which may lead to a problematic and erroneous result. The Kalman filter (KF) algorithm can be used to reduce these variations by adjusting the gain matrix based on the error between the model-observed value and the actual value of the terminal voltage.

Table 6 provides an overview of the Kalman filter for a LTI system and Fig. 5 illustrates recursive KF algorithm steps.

All of these will ensure that the SoC estimation approaches the true value. Observability of the linear system is investigated from the construction of the observability matrix O_b. The observability matrix O_b can be found by,

$$O_b = \begin{bmatrix} C & CA & CA^2 \end{bmatrix}^T \tag{5}$$

C and A values are given from the right-hand side elements of Eqs. (2) and (3) respectively. The parameter values originate from the parameter estimation step are inserted. Then the observability matrix is calculated in this article and the observability matrix is found to be full rank as per requirement of the observability criteria.

3.4.1. Choice of Kalman filter control parameters

Correct values of the process noise covariance matrix Q and measurement noise covariance R play a vital role for obtaining a fast and reasonably accurate tracking of battery terminal voltage. The covariance data give the algorithm information on the amount of variation to be expected on each state variable as well as possible dependency between several state variables. The initial covariance matrix P_0 is chosen to be,

$$P_0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{6}$$

By attributing a large covariance to a certain state, i.e. 1, the design should convey the algorithm what magnitude of variation it can expect for this variable. Initialization of the covariance matrix describing disturbances on the system Q is complicated by the fact that knowledge of model inaccuracies and system disturbances is limited, particularly as each cell has different characteristics [25]. During the measurement and time updates, the algorithm recalculates the state covariance matrix. The time update predicts the covariance matrix for the next sample period, given its value at the present period, based on the model equations. The process covariance Q allows adapting the individual covariance elements. The measurement update recalculates the covariance matrix based on the difference between the battery terminal voltage predicted by KF algorithm and the measured counterpart. Since no strict stability and tuning methods are available for initializing the KF, recourse to empirical tuning is normally adopted. Information about the system noise contribution is contained in matrices Q and R, determining the accuracy of the filter's performance, since they mutually determine the action of the KF gain matrix and estimation error covariance matrix representing measurement noise. A prudent choice of Q is, therefore, obtained from empirical studies under the simplifying assumption that there is no correlation between the elements of σ_k and the noise present on each cell's voltage transducer, thereby leading to a diagonal Q. The choice of a pure diagonal process noise matrix ensures maximum insensitivity for the initial errors.

$$Q = \begin{pmatrix} 0.00001 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{7}$$

The noise covariance R is also an important parameter, indicating the accuracy and noise sensitivity of the measured data. R can be estimated from knowledge of the battery terminal voltage.

Table 6
Kalman filter derivation from continuous time-invariant linear system in state-variable form.

Continuous time-invariant linear system in state-variable form		
State equation	$\dot{x}(t) = Ax(t) + Bu(t)$	Nomenclature: $u \rightarrow$ Applied input vectors; $x \rightarrow$ State vectors; $y \rightarrow$ Measured output vectors; $A \rightarrow$ Dynamics matrix; $B \rightarrow$ Input matrix;
Output equation	$y(t) = Cx(t)$	$C \rightarrow$ Measurement matrix.

As the system components A, B, C form a linear time invariant (LTI) system, meaning that while an implementation of the linear Kalman filter. Assuming the applied input u is constant during each sampling interval, a discrete-time equivalent model of that system is given by,

Discrete form: Where $A_d \approx I + AT_c$; $B_d = BT_c$; $C_d = C$, and T_c is the sampling period.

$x_{k+1} = A_d x_k + B_d u_k$
$y_{k+1} = C_d x_k$

To accommodate the uncertainties, the system is now assumed to be corrupted by stationary Gaussian white noise, via the additive vectors σ_k and μ_k—the former being used to represent system disturbances and model inaccuracies and the latter representing the effects of measurement noise. Both σ_k and μ_k are considered to have a zero mean value, for all k, with the following covariance matrices (E denotes the expectation operator)

$E[\sigma_k \sigma_k^T] = Q$ for all k; $E[\mu_k \mu_k^T] = R$ for all k

The resulting system is, therefore, described by the following expressions	Where, $z \rightarrow$ Vector of measured outputs after being corrupted by noise; $\Gamma \rightarrow$ Coupling matrix that governs the influence of

$x_{k+1} = A_d x_k + B_d u_k + \Gamma \sigma_k$
$z_{k+1} = C_d x_{k+1} + \mu_{k+1}$

the disturbance/modelling inputs on each state and is shown diagrammatically in Fig. 5. For notational purposes, $\hat{x}_{(k+1/k+1)}$ represents an estimate of x at step $k+1$ based on all the information including time step $k + 1$.

The cost function of the optimisation of Kalman filter is given by,
$\min\{E[(x_{k+1} - \hat{x}_{k+1/k+1})(x_{k+1} - \hat{x}_{k+1/k+1})^T]\}$

A property of the KF is that the estimated state vector $\hat{x}_{(k+1/k+1)}$ of the system, at time $k + 1$, minimizes the sum-of-squared errors between the actual and estimated states, meaning that for a recursive implementation, the KF estimate $\hat{x}_{k+1/k+1}$ is calculated from the previous state estimate $\hat{x}_{k/k}$ the input u and the measurement signals. The available input/output data at each sample step is, therefore, considered to be, $u_1, u_2, \ldots, u_k$, and u_{k+1} and $z_0, z_1, z_2, \ldots, z_k$, and z_{k+1} [9].
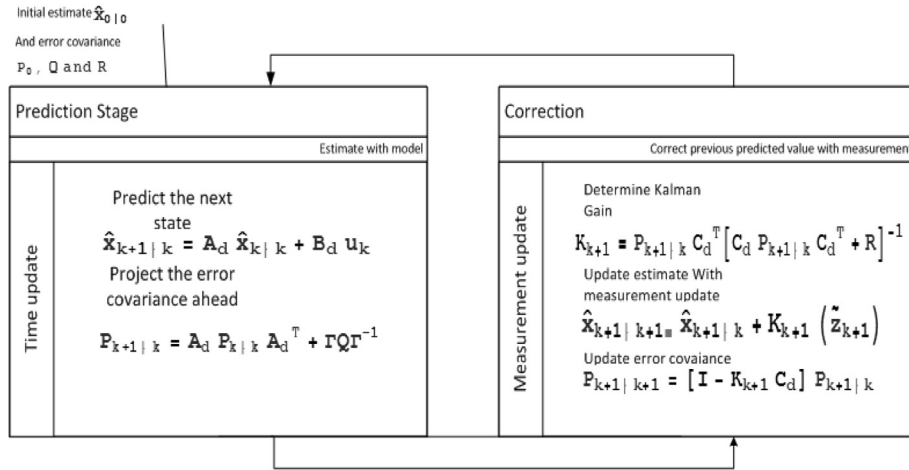
Fig. 5. Illustration of recursive KF algorithm steps.

The variance is obtained from the square of the root-mean-square (rms) noise on each cell, and is assumed to be Gaussian distributed and independent [9]. The ratio between the process noise covariance Q and the measurement noise covariance R is of significant importance as well. It determines whether the algorithm puts its trust for updating the state and its covariance in the measurement or rather in the state space model. Due to uncertainty on the model, it is important to attach sufficient weight to measurement data. Careful selection of R can provide estimates with enhanced noise immunity [9,25]. It is also needed to be taken into account that a large covariance offer the algorithm to adapt the value of certain state variable, but this might lead to a large overshoot and increase settling time as well. In order to obtain an optimal estimator, the noise covariance matrices should be as close to the actual noise parameters as possible. In case of a non-conformity exists between the battery terminal voltage stemming from the model and the actual measurement, the algorithm assesses which states have the highest covariance value and thus are most likely to be the cause of the deviation. Taking all of the above facts into account, the noise covariance R is ultimately therefore chosen to be,

$$R = 0.25 \tag{8}$$

3.5. Hysteresis correction with modified OCV graph

The hysteresis phenomenon is a common place for most of the battery cell chemistries. There is no scope of the used model to remedy the problem by itself. So a correction mechanism should be employed. The battery cells demonstrate the hysteresis effect as an open circuit voltage path that is not identical for the charge and discharge of the battery at different states of charge. Generally, the OCV after previous charge is higher than the OCV after discharge at the same SoC value. In case of the switching between the charging and discharging curves (due to imposition of charge and discharge pulse) makes the state of charge estimation problematic since switching from the OCV leads to abrupt changes in the output SOC what physically is not coherent for a battery cell. To find the real hysteresis a very low rate charging and discharging test at $I_t/100$ is proposed to be performed on the battery under test. Because the charging and discharging rate is very small and it is assumed that the measured voltage is the approximation of true OCV. This small current is now used as input for the OCV voltage determination through the usage of Kalman filter. The output of Kalman filter Uo is assumed to be the modified OCV of the battery cell under test. Thereafter, the estimated bulk voltage (Uo) is assigned to get the respective state of charge at different modified OCV level. From now, the OCV voltage is assumed to be the modified OCV voltage level. In case the current through the cell is changed from charge to discharge, the respective voltage is also changed. To smoothen the transition, a probabilistic control parameter is introduced. The probabilistic control parameter is established to regulate the transition between charge and discharge graph and the parameter is named as *Confidence Level*, CL. It represents the probability of a battery being on charge state (state 1) or discharge state (state 0) or intermediate states between charge and discharge (0 < state<1).

The charging state depends on the occurrence of the consecutive charge pulse amount which can be controlled by adjusting the amount of consecutive charge pulses through a predefined parameter *Transition time for charge state*. The discharge state can be found using the replication of the same method by the predefined parameter *Transition time for discharge state*. *Transition time for charge or discharge state* are chosen as parameters to accommodate the smooth transition between the discharge and charge OCV curve. In this article, they are given as 60 s based on empirical observation. So, to change from charge OCV to discharge OCV (or vice versa) takes at least 60 s and in between intermediate values will be taken, as explained hereafter. X_{ctrl} is the variable that determines the confidence level. After initialization this control parameter acts as a probabilistic counter that adds or subtracts the normalized sampling time by either transition time for discharge or transition time for charge.

The control parameter X_{ctrl} is governed by the following equations,

$$X_{ctrl}(i) = \max\left(X_{ctrl}(i-1) - \frac{\text{Time difference between two consecutive sample}}{\text{transition time for discharge}}, 0\right) \tag{9}$$

$$X_{\text{ctrl}} = \min\left(X_{\text{ctrl}}(i-1)\right.$$
$$\left.+ \frac{\text{Time difference between two consecutive sample}}{\text{transition time for charge}}, 1\right)$$
$$(10)$$

$X_{\text{ctrl}}(0) = 0$; When the first pulse is found to be discharge pulse or idle state (zero current).

$X_{\text{ctrl}}(0) = 1$; When the first pulse is found to be charge pulse.

It is important to define the direction of the current. In this particular case negative current is assumed to be the discharge pulse and positive current is assumed to be charge pulse.

Obviously, the introduction of this control parameter ensures the smooth transition between the curves. This X_{ctrl} parameter works like a weighting factor ($0 \leq X_{\text{ctrl}} \leq 1$) for determining the intermediate voltage level in between the transition of two curves. The *Confidence Level* is a probabilistic parameter value that takes the shape of an error function depending on the control parameter, X_{ctrl}. erf(z) is the "error function" encountered in integrating the normal distribution (which is a normalised form of the Gaussian function) [26]. It is defined by,

$$\text{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt \qquad (11)$$

The *Confidence Level*, *CL*, is the error function with z replaced by X_{ctrl} in Eq. (11).

Erf has the values erf(0) = 0 and erf(∞) = 1.

So, the intermediate voltage level is to be determined by using the following formula,

$$\text{Intermediate voltage} = \text{CL*OCV}_{\text{charge}} + (1 - \text{CL})\text{*OCV}_{\text{discharge}} \qquad (12)$$

This intermediate voltage is the OCV level which is used to find the SoC level of the battery. The *Confidence Level* parameter functions as a non-linear smoother as it take care of all the abrupt variations that is coming from the transition of charging–discharging OCV–SoC curves. Depending on the consequence of the hysteresis effect, the method described could be adjusted for different batteries making the systems suitable for diverse chemistry battery systems where hysteresis effects are predominant (e.g. NiMH batteries and others).

4. Result and discussion

4.1. Parameterisation

Initial parameters required for the battery RC model are obtained from the offline analysis of IPT characterisation data using the chronological combination of the modified Genetic Algorithm and the modified Levenberg Marquardt method. The fitted voltage using the estimated parameter sets are reasonable with respect to the battery test data. Most of the simulated and measured results lined up very closely and this is illustrated in Fig. 6. Although some discrepancy exists between the measured and modelled voltages, the underlying dynamic characteristics are essentially the same, with the principle difference being the voltage response between IPT pulses. But the most important observation is that the model is able to track the dynamics of the system. Most of the simulated and measured results line up very closely. Because of the usage of the two level parameter estimation system guarantees a more optimised version of the parameters, the paradigm has been used. The selection of the initial parameters and constraints for optimisation is not a trivial process. Determining which parameters affected only the charge–discharge cycles is not straightforward. The parameter values are chosen only if the curve fitting is considered to be reasonable. Weight coefficients (β and n in Eq. (4)) are used to track the short duration peaks to include in the parameter estimation scheme when using the model, because the raw current and measured voltage values have some magnitude difference especially on different current rates. The effect of the introduction of the scheme is obvious in Fig. 4. In order to comprehend the steps of modified Genetic Algorithm that has 120 population and 80 generations, different arbitrary steps exhibiting the evolution of parameters of the modified Genetic Algorithm are
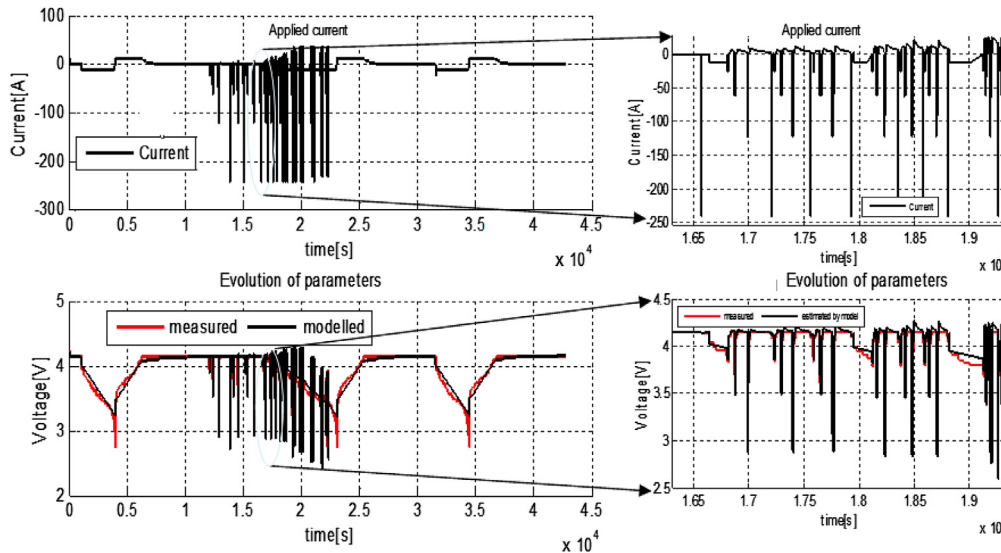

Fig. 6. Different steps of parameter estimation. Applied current [top] and measured vs. modelled voltage [bottom] and right side figure gives a closer look to selected portion. Measured Voltage in Red and fitted by model voltage in Black. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
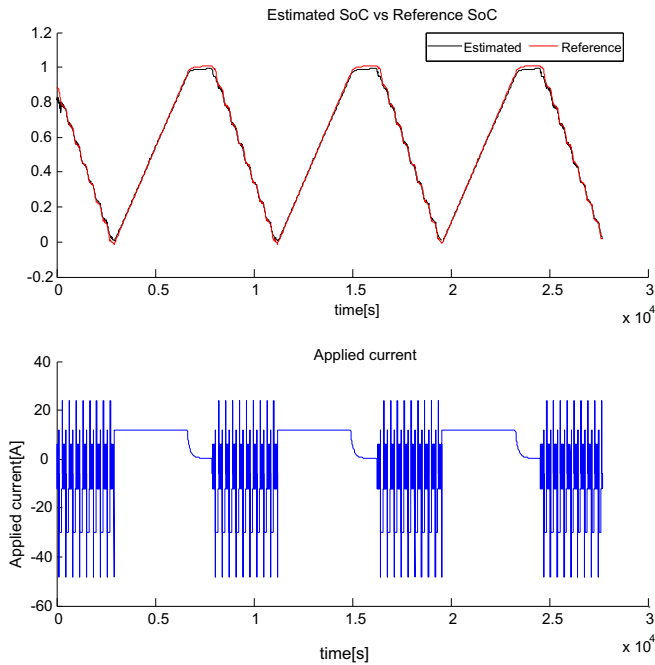
Fig. 7. [Top] Illustration of comparison of measured and estimated state of charge [bottom] applied current.

shown in Table 3. In the same way, the evolution of the battery cell parameters using modified Levenberg Marquardt method is shown in Table 5.

The similar experiment is repeated on the same cell three times and also the execution of the estimation algorithms are done three times making nine combinations for the test cell. Simple averaging of these combinations is taken as the possible parameter set. An indication of the output after execution of modified Genetic Algorithm can be found in Table 4.

The optimised best population is to be inserted as initial search position for modified Levenberg Marquardt method. The parameter sets, obtained after execution of modified Levenberg Marquardt algorithm, can also be seen in Table 4.

The optimised parameter sets, that are the output of the algorithm, are assumed to be the accurate one. These parameters are used as fixed parameters for the state estimation of the battery. The computations are carried out on a processor Intel® Core™2 Duo CPU P8600 at 2.4 GHz clock speed with 4 GB RAM and the operating
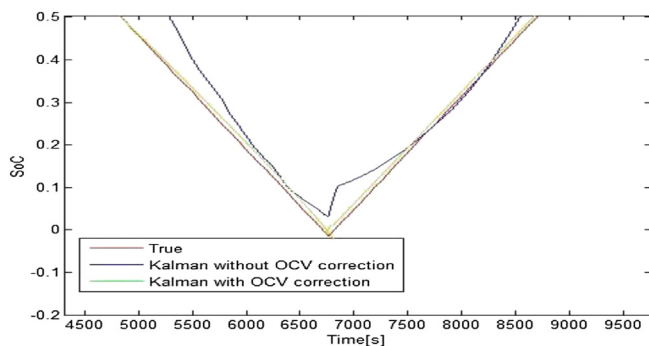

Fig. 8. The figure proves the necessity of hysteresis compensation using SoC OCV curve. In case of no hysteresis presents (in blue graph) the graph may bend and jump into abrupt changes of SoC The usage of OCV correction will mitigate the problem (green curve). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
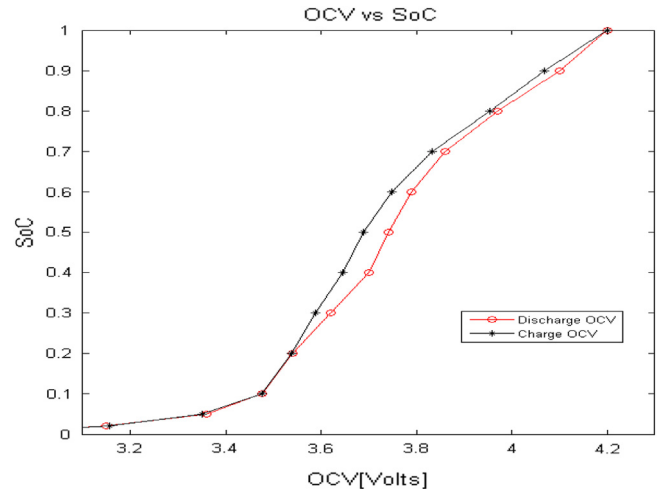

Fig. 9. Modified OCV vs. SoC graph.

system is Windows 7 Enterprise edition. The codes for both methods were self-devised and hand-optimised in Matlab environment (MATLAB® Version 7.13.0.564, The Math Works, Inc., Natick, MA, USA).

4.2. State estimation

The model-based SoC estimation greatly depends on the model selection, parameter estimation algorithms and without proper control this may lead to a fluctuant, even divergent result. For this reason, the stochastic principles behind Kalman filters are very attractive as the filter finds the optimum between the model prediction and the measured data. A statistical predictor-corrector structure thereby offers obvious advantages over other alternatives. Only terminal quantities of the battery can be measured, i.e., the input current and the measured terminal voltage [11]. The confirmation of the fact that the estimated SoC is tracking the actual SoC is illustrated in Fig. 7. In this article a load profile is used as it is specified in the IEC 62660-1 standard for BEVs [13] with a modification to accommodate the full charging and discharging of the full battery cell. Observing the load profile to verify the SoC
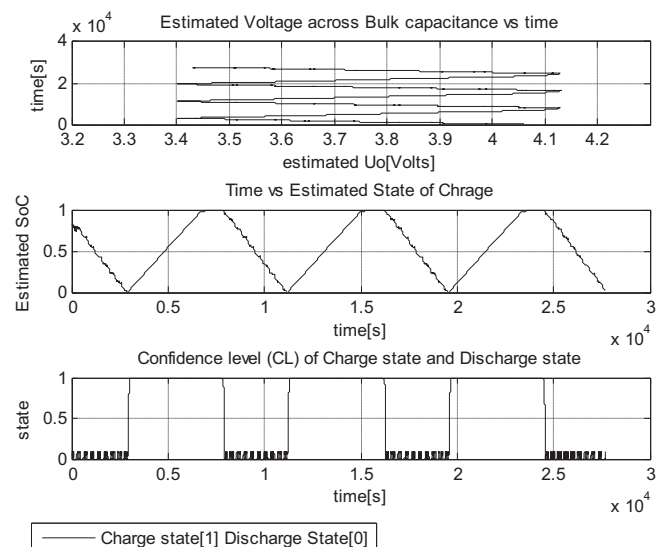

Fig. 10. Estimated OCV [top], state of charge [middle], Confidence Level [bottom].
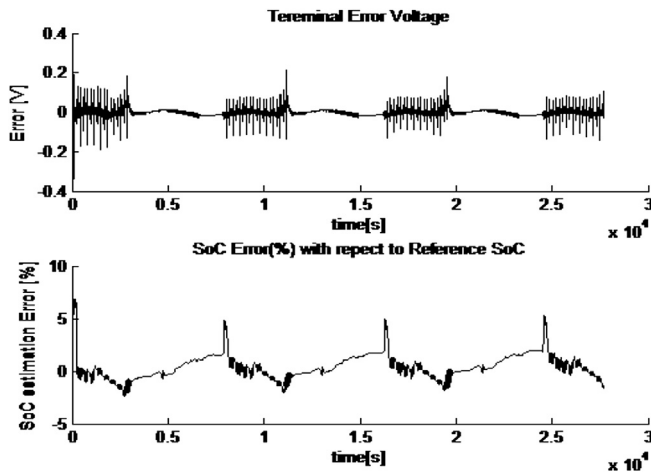
Fig. 11. Terminal error voltage [top right] and SoC [%] error [bottom right].

estimation approach and the sampled current profiles are shown in Fig. 7, the subjected battery to the BEV load profile consist of two modes: a discharge-rich phase where the SoC depletes to 0% (empty cell), and a charge-rich phase where the SoC rises again to 100% (full charged battery cell). Measurements using this profile are used to compare the performance of conventional integration-based methods for estimating SoC with those predicted from the presented state estimation schemes. The experimental reference SoC used in Fig. 7 is based on the *PEC SBT0550* test bench results after proper adjustments as follows:

The capacity of the battery is acquired from the constant discharge at a rate of $0.01I_t$. In order to get the true SoC by an experimental approach, firstly, the battery cells are charged by constant current and constant voltage charging until the current drops below $0.01I_t$. This is defined as an initial SoC being 1.0. Since the true values of the initial SoC and the terminal SoC are determined, the simple coulomb (Ah) counting method is used to calculate the experimental SoC based on the load current profile, which is calculated based on the true values of the initial SoC and terminal SoC, is applied during the calculation. The effectiveness of the modified OCV curve and the hysteresis approach are shown in Fig. 8. The bump in SOC that would be visible if there is a change from discharge to charge, has disappeared. The output of Kalman filter U_o is assumed to be modified OCV of the battery cell under test. The modified OCV level is depicted in Fig. 9, plotted over SoC. Fig. 9 serve as a look up table for finding OCV—SoC relationship.

Thereafter, the estimated bulk voltage (U_o) is assigned to get the respective state of charge at different modified OCV level. Because of the introduction of this confidence level parameter it avoids computationally expensive EKF as proposed by Ref. [10] to offset the error instigated by hysteresis effect. This parameter also functions as a smoother as it take care of all the abrupt variations. State of charge estimation is found only through the computation of the modified OCV. The voltage distribution over the time—this modified OCV voltage is used in co-ordination with hysteresis curve to determine the consequent SoC curve as shown in Fig. 10. The terminal error voltage and estimated SoC [%] error is plotted in Fig. 11. The state of charge error is greater on preliminary stage but after sometime the algorithm converges to acceptable error.

5. Conclusions

The usage of two nonlinear optimisation algorithms offers the best fit to predict the dynamic effects of the battery cells. Best dynamic performance of any particular model gives a more accurate SoC estimation. The article proposes a unique combination of Maximum Likelihood (modified Genetic Algorithm and modified Levenberg Marquardt) and Expectation Maximisation (Kalman filter) in coordination with a statistical based approach to offset hysteresis effect in order to determine state of charge of the cells in a battery pack. The estimation framework is online executable while finding accurate SoC within acceptable error limit. The framework is found to be robust, flexible, modular, and executable in a time constrained environment as the battery parameters are calculated offline and assumed to be constant for the estimation of state of charge. The system can be coupled with an existing battery management system (BMS) with an additional digital signal processing (DSP) board implementation. Nevertheless, undesired ageing effects can occur when test batteries are charged and discharged repeatedly which correspond to different parameter values signifying different State of Health (SoH) level. In this article, a new battery cell is tested for only limited number of discharge and charge cycles to minimize ageing effects. The future work is on the integration of ageing and temperature effect into the state of charge framework while maintaining robustness of the system. This will lead to an extra dimension for the parameter tables but does not change the presented methodology. It is notable that while the presented work has focused on a specific battery model with a system identification approach, with a novel hysteresis compensation scheme that makes the framework reusable for different cell chemistries. Ultimately, due to the all presented benefits, it is expected that similar state-of-charge scheme can be adopted for different cell chemistries with minimal changes on the algorithm structure. Also, depending on the effect of the hysteresis, the parameters could be adjusted for the particular battery chemistry making the system suitable for different chemistry battery cells. Overall this method within state of charge estimation method that exists today is best suited for varying dynamic conditions of the bigger multiple cell battery pack e.g. HEV, BEV, grid scale battery pack in a time constrained environment.

References

[1] D. Linden, T.B. Reddy, Handbook of Batteries, third ed., McGraw-Hill, 2002.
[2] N.A. Windarko, J. Choi, Hysteresis modeling for estimation of state-of-charge in NiMH battery based on improved Takacs model, in: Intelec 09—31st International Telecommunications Energy Conference, IEEE, New York, 2009, pp. 598—603.
[3] M.A. Roscher, D.U. Sauer, J. Power Sources 196 (2011) 331—336.
[4] S. Piller, M. Perrin, A. Jossen, J. Power Sources 96 (2001) 113—120.
[5] V. Pop, H.J. Bergveld, P.H.L. Notten, J. Veld, P.P.L. Regien, Measurement 42 (2009) 1131—1138.
[6] I.S. Kim, IEEE Trans. Power Electron 23 (2008) 2027—2034.
[7] D.V. Do, C. Forgez, K.E. Benkara, G. Friedrich, IEEE Trans. Veh. Technol. 58 (2009) 3930—3937.
[8] H. He, X. Zhang, R. Xiong, Y. Xu, H. Guo, Energy 39 (2012) 310—318.
[9] B.S. Bhangu, P. Bentley, D.A. Stone, C.M. Bingham, IEEE Trans. Veh. Technol. 54 (2005) 783—794.
[10] A. Vasebi, M. Partovibakhsh, S.M.T. Bathaee, J. Power Sources 174 (2007) 30—40.
[11] G.L. Plett, J. Power Sources 134 (2004) 252—261.
[12] V.H. Johnson, A.A. Pesaran, in: 17th Annual Electric Vehicle Symposium, Montreal, Canada, 2000.
[13] G. Mulder, N. Omar, S. Pauwels, F. Leemans, B. Verbrugge, W. De Nijs, P. Van den Bossche, D. Six, J. Van Mierlo, J. Power Sources 196 (2011) 10079—10087.
[14] H. He, R. Xiong, J. Fan, Energies 4 (2011) 582—598.
[15] M. Coleman, C.K. Lee, C. Zhu, W.G. Hurley, IEEE Trans. Ind. Electron. 54 (2007) 2550—2557.
[16] I.E. Commission, in: International Electrotechnical Commission, 2010.
[17] F.U.S.D.o, Energy, FreedomCAR Battery Test Manual for Power Assist Hybrid Electric Vehicles, 2003.
[18] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.
[19] A.J. Jones, Nature 363 (1993) 222.
[20] D. Corne, M. Dorigo, F. Glover, D. Dasgupta, P. Moscato, R. Poli, K.V. Price, in: McGraw-Hill Ltd., UK, 1999.
[21] D. Ackley, A Connectionist Machine for Genetic Hillclimbing, Springer, 1987.

[22] G.H. Golub, C.F.V. Loan, Matrix Computations, third ed., Johns Hopkins University Press, 1996.
[23] M. Lampton, Computers in Physics, vol. 11, 1997, pp. 110–115.
[24] M.I.A. Lourakis, in: 2005.
[25] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics (Intelligent Robotics and Autonomous Agents), The MIT Press, 2005.
[26] F.S. Acton, Numerical Methods That Work, second ed., Math. Assoc. Amer., Washington, DC, 1990.

Glossary

DOD: depth of discharge (%)
BEV: battery electric vehicle
HEV: hybrid electric vehicle
HPPC: hybrid pulse power characterisation test
IPT: integrated pulse test
NMC: lithium-ion nickel cobalt manganese oxide cell
OCV: open circuit voltage
RT: room temperature
SoC: state of charge (%)
KF: Kalman filter
GA: Genetic Algorithm
LM: Levenberg Marquardt
Ah: ampere hour
DSP: digital signal processing
BMS: battery management system
EV: electric vehicle